(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
29 November 2001 (29.11.2001)

PCT

(10) International Publication Number
WO 01/90836 A2

(51) International Patent Classification[7]: G06F

(21) International Application Number: PCT/IL01/00472

(22) International Filing Date: 23 May 2001 (23.05.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/577,619 25 May 2000 (25.05.2000) US

(71) Applicant *(for all designated States except US)*: **EXTENT LDT** [IL/IL]; 28 Betzalel st, 52521 Ramat Gan (IL).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: **GARINI, Jonathan** [IL/US]; 2542 Brenton Point DR. Reston, Reston, VA 20191 (US).

(74) Agent: **NAOMI ASSIA LAW OFFICES**; P.O.B. 314, 54 Sokolov ST., 47103 Ramat Hasharon (IL).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

WO 01/90836 A2

(54) Title: REAL-TIME (ALL THE TIME) OPERATION SUPPORT SYSTEM AND METHOD

(57) Abstract: The present invention provides an operations support system and method for communications services that compiles exact usage data in real-time. In one embodiment, the present invention is a billing system for IP-based communications. The system employs a server and associated technology to maintain the state of all active services as a function of time. The server software is providing the value of a parameter at any given moment in time by storing it as a function of current time. The server serves as the point of authority for current user information during the session. At the end of the session the information is stored in a database for archival, query and batch processing. The method activates events according to predetermined business rules. Such system is accomplished by predicting events that may occur in the future through maintenance of a dynamic updated table of events for future business.

REAL-TIME (ALL THE TIME) OPERATION SUPPORT SYSTEM AND METHOD

BACKGROUND OF THE INVENTION

1. Field of the Invention

The invention relates to an operation support system (OSS) infra-structure and method for meeting the needs of various types of service providers. More specifically, the invention provides a true real-time support system and method that meets such needs as usage collection, provisioning, order entry, subscriber management, customer care and rating/billing requirements for service providers operating on an IP platform.

2. Description of Related Art

The appearance of next generation, IP-based communication services and devices has become virtually an everyday event. With the charged atmosphere of the Internet, and the subsequent creation of the new economy, companies seek to develop new services that give them the edge and, ultimately, enable them to profit from the opportunities the IP platform creates. Service providers therefore require a support system infrastructure that effectively meets their usage collection, provisioning, order entry, subscriber management, customer care and rating and billing requirements. Given the pace of the Internet, such an infrastructure should be dynamic, and enable providers to instantly deploy and modify hosted services.

To address the above needs, support systems strive to synchronize usage data in real-time -- meaning that the billing and complementary support systems can determine the exact state of usage across all subscribers (and corresponding account balances) instantly, at any moment in time, and that this information can be instantly accessed by

any support system or subscriber. (For background on real-time billing, refer to "Batch Systems for Internet Billing? Think Real-time." By Dr. Matthew Lucas and Dave Labuda, Billing World, January 1999.)

The importance of real-time data in the service provider environment can be understood on three basic levels, each of which provides a compelling reason for the availability of real-time data.

First, real-time systems satisfy customers. People are used to getting real-time data in other venues. For example, when customers go to their bank on line, they see their current balance – even if they made a deposit just moments ago. Likewise, when they go to an on-line commerce site such as Amazon.com, they will know the exact status of their purchase order. When they want to check on a package that they shipped via FedEx, they will see its exact location. When they go to their communications service provider, however, they often receive dated information, information that is only updated when the service provider is able to update it. In addition to real-time account balance, today's customers also expect a real-time accounting of the services and operations they have accessed and utilized (e.g., multi-service CDR reports), when and for how long, for both themselves and every member of their account. Looking even further, customers will want on-line self-provisioning that takes affect immediately, instant fulfillment of services ordered, and on-the-spot cancellation of specified services.

A second valuable component of true real-time operation is that it can be used as a tool to enforce business rules, including pre-paid rules, as well as for fraud prevention. For example, a customer with a $100 credit limit will not reach a balance of $102.75 - almost the defined limit, simply because the OSS didn't know when to disconnect him.

Currently, both batch and transaction based real-time systems do not solve the problem of accurate credit limits and similar business rules.

Thirdly, true real-time operation can help a provider market its services. Since real-time provides a powerful window into the subscribers' world, the provider can develop and promote relevant incentives and packages. For example, real-time, live promotional item offers immediacy and can help increase customer loyalty and revenue-making capabilities.

In IT systems, features typically come at a price. That is particularly evident in engineering systems with time-based, performance constraints. In fact, guaranteeing instant response-time is one of the most difficult performance requirements for any system. Therefore, such solutions are expensive since they require more computing infrastructure engineered specifically to handle peak loads.

Various elements at the transport layer (such as voice, data and video) are steadily merging, as well as the services offered at the application layer (such as Web content, games servers, rented applications, rich media and video on demand). The result is that new business models are being developed to increase revenue via value-added application and content services. Some providers may offer free communications services while charging for value-added application and content services; others will opt to provide free content and application services while making their revenue from advertising. Vertical ISPs, free ISPs and MSOs are all examples of communications providers developing these business models. But these innovations can only be enabled through usage-based billing, which requires true real-time data.

There are a variety of billing systems currently in existence that claim to deliver a real-time system. These systems, however, fall short of providing an actual real-time system adequate to meet the needs of service providers and customers/users. These existing billing systems include the following.

1. First Generation - Batch-based systems. A first generation solution is characterized by processing batches of events at predetermined intervals. The billing function in this type of system often coincides with invoicing time. Some vendors have shortened this interval to hours to gain "near" real-time functionality.

2. Second Generation – Transaction (session)-based systems. This type of system is characteristically event-driven, the event typically being the commencement of a session. An entire billing process is usually carried out at the end of a session. This type of system knows, in real-time, when a session begins and ends, but does not update its information while the session is in progress. It only updates that retroactively upon session completion. Therefore, its real-time capacity is quite limited considering that sessions might continue for an extended time, during which a variety of business rules should be triggered.

When handling real-time data, first and second generation systems essentially provide a similar level of information resolution, which in many cases is a few hours off at best. In first generation systems, account resolution occurs at preset intervals. In second generation systems, the data is typically updated at the end of a session. So, in one sense, a transaction-based real-time system does operate in real-time. But, as illustrated in FIG. 1, since such systems do not provide accurate data at any given moment, they are merely "near" real-time. Also, although in a second generation system

it is possible to enforce a session limit during the authorization phase and work around the session's time limit, this workaround is not generally effective work in a multi-service, multi-account environment, where business rules may be affected by simultaneous sessions of converged services, family or corporate accounts. As is seen, there does not currently exist a true, real-time operations support system that adequately meets the needs of service providers.

SUMMARY OF THE INVENTION

In view of the above disadvantages of currently available systems and methods, service providers require an actual real-time billing system that (1) goes beyond the boundary of a session's start time, and reaches right into the session itself, keeping track of what actually occurs during a session – while it is happening, (2) updates the billing and customer care systems on demand, and (3) scales to large numbers of subscribers. Such a support system and method is not currently available today and is provided in the below-described invention.

In brief summary, it is an object of the present invention to provide a next-generation billing system that compiles exact usage data in real-time. To that end, the present invention provides a method on which distributed or non-distributed operations support system and employs a server and related technology (a "state cluster") to maintain the state of all active subscribers' services as a function of time. By holding the state of the active service and their vectors, as well as a formula that calculates a required parameter as a function of the current time, the system is always delivering real-time

data, without polling network elements and/or a RDBMS that can cause serious performance degradation and scalability problems.

It is also an object of the present invention to provide a system and method that activates a chain of events according to predetermined business rules based on the state of a given subscriber session. The present invention accomplishes this by attempting to predict events that may occur in the future through maintenance of a table of events that is triggered every second. The table is dynamically updated, as newer events might affect the timing of older events in the table. This method is far more efficient than to evaluate all business rules every second, an alternate method that can also cause serious performance degradation and scalability problems.

It is further an object of the present invention to provide a billing system that is scaleable to handle large numbers of transactions and users within a network, as described below.

DESCRIPTION OF THE DRAWINGS

FIG. 1 is a graphical representation of the comparison between batch, transaction and true real-time account balance availability.

FIG. 2 is a schematic diagram of a preferred embodiment of an operations support system in accordance with the invention.

FIG. 3 is a flow diagram of a preferred embodiment of a method in accordance with the present invention.

DESCRIPTION OF PREFERRED EMBODIMENTS

The present invention employs a server software and related technology (collectively a "state cluster") which processes usage data in actual real-time. Namely, rather than employing a simple system that services upwards of a million users and polling all of the system elements to determine the status of each subscriber and services each is accessing, the present invention uses a "state cluster" that maintains the state of all active subscribers' services as a function of the current time. The system always knows each and every subscriber's status. By holding the state of all active subscriber's service and their vectors (e.g. pricing and rating rates and rules), as well as a formula that calculates a required parameter (such as the account balance) as a function of the current time, the system can always deliver real-time data accurate up to the moment. Therefore, the system has uninterrupted access to that service parameter, without polling other elements of the system a single time. The set of formulas is dynamically updated as events that affect it occur. The system stores data on customers, groups of customers, entities and aggregates all of that real-time data in the "state cluster", as well as a dynamic list of events to be triggered according to defined business rules.

In addition to true real-time data, the invention also provides a mechanism that activates a chain of events according to predetermined business rules based on the state of a given session. This permits, for example, a service provider's disconnecting a subscriber in the middle of service the moment their credit runs out, or conversely, to give a bonus – on the spot - if a subscriber used a certain service more than a set amount of time. The number of business rules of this type required is obviously great. Conceivably, each of those rules would need to be run every second against all of the functions stored in the state server. Such would, however, put an unbearable strain on the

system. This predicament is overcome in the present invention by the system's attempting to predict events that may occur in the future. That is accomplished by maintaining a dynamic table of events that is checked every second. When the session starts the system calculates a rule. The system predicts the time a business rule will trigger and inserts the rule into a dynamic table of events with a predicted time, that is either accurate time or may be before the accurate time dependent on the complexity of the rule. When the time comes the system tests this rule again to see if it meets the rule requirements. If yes – it executes the rule action, if not it recalculates and inserts the rule back into the table of events. The table itself is also dynamically updated, as newer events might affect the timing of older events in the table.

More specifically, FIG. 2 shows a preferred embodiment of an operation support system 200 in accordance with the present invention. Shown at 201 are various types of information inputs to the system 200 such as "Content," "Wireless Application Protocol (WAP)," "Application," "Dialup," "Broadband," "Voice," "Video," and other types of information inputs. Inputs 201 are the network transaction information or data which result from system subscribers' activities. This transaction information 201 resides in various network devices and application servers such as routers, IP switches, and web servers, among others, from which it is collected by system 200. Each of the transaction information inputs 201 represent a different transaction input.

Mediation agent components 203 of system 200 collect the network transaction information 201. Each source of network transaction information 201 has its own dedicated mediation agent 203. Mediation agents 203 also perform the task of reformatting network transaction information 201 into a known format of the subsequent

system component, rating engine 209; mediation agent components 203 transmit the information 201 to rating engine 209. There are a few network resources capable of sending network transaction information 201 directly to the system via direct access component 205, which, in turn transfers the network transaction information 201 received to rating engine 209. In addition, there are others applications (e.g., (1) a payment is processed through a third party payment processing software or 2) a customer care representative credits the customer due to a dispute on a specific billing charge, or 3) legacy systems, such as accounting software, import/export data to/from the system) which are capable of accessing the system components directly.

Each of the network devices and application servers (from where information 201 is forwarded to system 200) needs to be configured to support the required services for each subscriber/user. Accordingly, provisioning engine 207 is dedicated for sending commands that enable or disable a service for a specific subscriber, and is programmed to be familiar with the network devices' APIs in order to perform that task.

Rating engine 209 receives the network information 201 from mediation agents 203, and calculates the amount this transaction is charged for. Rules engine 211 manages pre-defined directives for the system 200 to be triggered in a determined state (e.g., a two hour video stream is charged for 6$ - 3$/hour). These directives are supplied by an administrator of the system, and can include, disconnection of a low credit customer if he exceeds 100$ in the current cycle. Registration services program 213 is a web based application that allows prospective subscribers to register onto the system 200 and to purchase plans and products. Customer care and Self-care components 215 are web based applications that allow existing subscribers to perform various changes or updates

to their service offering or view their real-time balance or other parameters status. All of these tasks are made on-line and executed in real-time. The customer service department of the system 200 also uses customer care application 215 (with higher permissions) in order to service user's requests. Payment authorization program 217 enables authentication and authorization of credit card payments in order to validate payments as close as possible to the time of the services ordering. Authentication services program 219 authenticates the use of a specific service for a specific user/subscriber. In addition, authentication services program 219 makes the decision whether a user has the required credit to perform a specific task or use a specific service. Enforcement services program 221 performs real-time actions against the network elements, such as disconnecting a user in case of zero credit. Although enforcement services program 221 resembles provisioning engine 207, enforcement services program 221 is executed while the user session is active.

Invoicing program 225 is responsible for the billing process which aggregates all rated transactions, periodic charges (recurring charges) and one-time charges which occurred in a specified period. The final output of the invoicing program 225 are bills to be paid by the subscribers. Accounts receivable program 227 handles payments, adjustments, refunds, transferring amounts between bills and resolving disputes. Offline reports program 229 provides the administrator of the system 200 with the opportunity to create and print periodic reports on financial data, system data, and actions which have occurred in the system.

Relational database management system (RDBMS) 232 holds the application and reference data of the system 200 for archival, query and batch processing. For example,

reference data resides in static entities which are not changed during usage processing including: customer details, pricing objects, invoicing templates, etc. On the other hand, application data is accumulated during the system usage processing and includes customer rated sessions, produced invoices, etc. Database 232 is in electronic communication with invoicing program 225, accounts receivable program 227 and offline reports program 229. Those programs provide updates to database 232 and query database 232 (at predetermined intervals) for the information required to perform the invoicing, accounts receivable and reporting functions. The engines and programs of layer 240 also provide updates to and queries of database 232 as described above.

System "state cluster" 235 has two components which facilitate real-time billing, account current status component 234 and the dynamic table of events 236. Specifically, while a user has an active session, account current status component 234 holds (stores) the user's identifying parameters as well as the service rating parameters. Together with the dynamic table of events 236, described below, the state server 235 acts as the point of authority for the user data instead of the customer database 232.

Dynamic table of events 236 stores a set of selected rules for users who have a session active. Rules are added to this component, being taken from rules engine 211, based on a mathematical calculation on the rules parameters, the session characteristics, the account current status and the user rating parameters. When rule requirements are met and a rule is triggered, the current state of the session is tested to determine whether it meets the rule requirements. Specifically, if the state meets the rule requirements, the rule action will be triggered. If not, the rule will be recalculated and inserted back into

the table 236. Rules specific for a customer are deleted from the table 236 when the session ends.

FIG. 3 shows the data flow performed by the state server 300. Specifically, a subscriber session starts, step 301, and triggers the action of state server 300, the state server now becoming the point of authority for the user between the session start time 301 and the session end time 315. In response to the session start 301, the state server 300 creates a current account status 303. State server 300 queries database 332 for the user details and rating parameters in order to update account current state table 334. In addition, all user's rules are fetched, step 305, and a calculation is made, step 307, for each and every rule. Rules calculation 307 predicts the rules triggering time according to the session parameters, the user's rating parameters and the rule itself. This calculation derives a record in the dynamic table of events 336 with a future predicted time to be executed.

Timer 309 represents a system event that occurs when a rule triggering time comes. When such an event occurs, the triggered rule is taken from the dynamic table of events 336 and a rule evaluation, step 311, is performed. During rule evaluation, one of two scenarios may occur. The first scenario is that the evaluated rule terms do not meet the rule requirements. In this case, the rule time is calculated again and state server 300 updates dynamic table of events 336 with the new time. The second scenario is that the rule requirements are met. Only then is the rule triggered and state server 300 sends an event, step 313, to the system in order to execute the rule. The last step in this case is that the rule is then deleted from the dynamic table of events 336.

When the session ends, step 315, the state server task is terminated for the user who made the session (unless another session is opened for this user). The session end, step 315, terminates, at step 317, the user's account current status records in table 334 and also terminates the rules, step 319, from dynamic table of events 336. This action calls off the state server from being the state of authority for the user and transfers the point of authority to database 332.

Additionally, there are cases in which a rule is added to the system, step 321, while state server 300 is the point of authority for a user status. In this case, the added rule must be calculated, step 307, and entered in the dynamic table of events 336.

What is claimed:

1. A method for providing a communications system access to real-time information, comprising the steps of:

a) employing a server as a single point of authority of access to information for at least one element of said system;

b) storing at least one parameter relating to said information as a function of real time information updates;

c) updating said function when an event that affects said parameter occurs.

2. A method according to claim 1 also comprising the steps of:

- predicting at least one future time depended on said function.
- storing said predicted future times.
- updating said predicted future time when an event that affects said future time occurs.

3. The method according to claim 2 also comprising the step of checking respective function at predicted time according to current information updates.

4. The method according to claim 3 also comprising the step of triggering at said predicted future times;

5. A method according to claim 1 wherein the parameter is an account balance.

6. A method according to claim 1 wherein the system is capable of providing the value of said parameter at any given moment in time.

7. The method of claim 1 wherein the system operations on a distributed computer network.

8. The method of claim 7 wherein the network is the Internet.

9. The method of claim 1 wherein the function is a business rules.

10. The method of claim 9 also comprising the step of evaluating said business rule according to information updates at predicted times to determine whether the rule requirements are met.

11. A method according to claim 9 in which a business rule is added to the system while the server is the point of authority.

12. The method of claim 10 wherein if the evaluating step determines that the business rule requirements are not met, a new time is calculated and the system is updated with the new time.

13. The method of claim 10 wherein if the evaluating step determines that the rule requirements are met, the rule is executed.

14. The method of claim 1 wherein the step of employing the server as a single point of authority is accomplished by software.

15. A method for managing a communications services system operating on a network comprising the steps of:

a) starting a user session;

b) triggering the action of a state server so that the server becomes the point of authority for current user information during the user session;

c) the server's processing user information;

d) ending the user session; and

e) once the user session has ended, terminating the state server as the point of authority for the current user information and transferring said point of authority to a storage device.

16. The method of claim 15 wherein the communication services system is a communications billing system.

17. The method of claim 15 wherein the network is the Internet.

18. The method of claim 15 wherein the communication services system provides Internet protocol based services.

19. The method of claim 15 wherein the storage device is a relational database.

20. The method of claim 15 wherein the step of the server's processing user information comprises the server's creating a user current account status record.

21. The method of claim 15 wherein the step of the server's processing user information also comprises the server's updating the user's current account state by querying the storage device for user information and rating parameters.

22. The method of claim 15 also comprising the step of storing at least one business rule associated with the user session.

23. The method of claim 15 further comprising the step of predicting the time the business rule will trigger.

24. The method of claim 23 further comprising the step updating predicted times according to new events.

25. The method of claim 23 further comprising the step of evaluating the business rules at predicted time according to current information updated.

26. The method of claim 16 wherein the step of predicting the time the business rule will trigger is accomplished according to user rating parameters.

27. A method according to claim 24 in which when the session ends, the user's account current status record is terminated.

28. A real-time communications operations support system connected to a network comprising:

(a) at least one network information input;

(b) an information receiving component which collects the network information input and is capable of transmitting said input;

(c) a server in electronic communication with the information receiving component wherein the server is capable of serving as the point of authority for processing current user information during a user session;

(d) a database which stores and retrieves user information, is electronically connected to the information receiving component, and is capable of taking over the point of authority from the server once the user session ends.

29. The system of claim 28 wherein the server stores information relating to users, and a dynamic list of events to be triggered according to defined business rules.

30. The system of claim 28 wherein the server comprises an account current status component which holds user parameters and service parameters, and a table of events which holds at least one business rule for a user who has a session active.

31. A method according to claim 30 wherein the table of events is capable of being dynamically updated.

32. A system according to claim 28 wherein the system is a communications billing system operating on an IP platform.

33. A system according to claim 28 wherein the network is the Internet.

34. A system according to claim 28 also comprising an invoicing component electronically accessible by the database and running a process that generates user bills.

35. The system of claim 34 which also comprises an accounts receivable component electronically accessible by the database and running a process that handles payment and adjustments to said bills.

36. The system of claim 28 also comprising a rating engine which receives information from the information receiving component and transmits said information to system services.

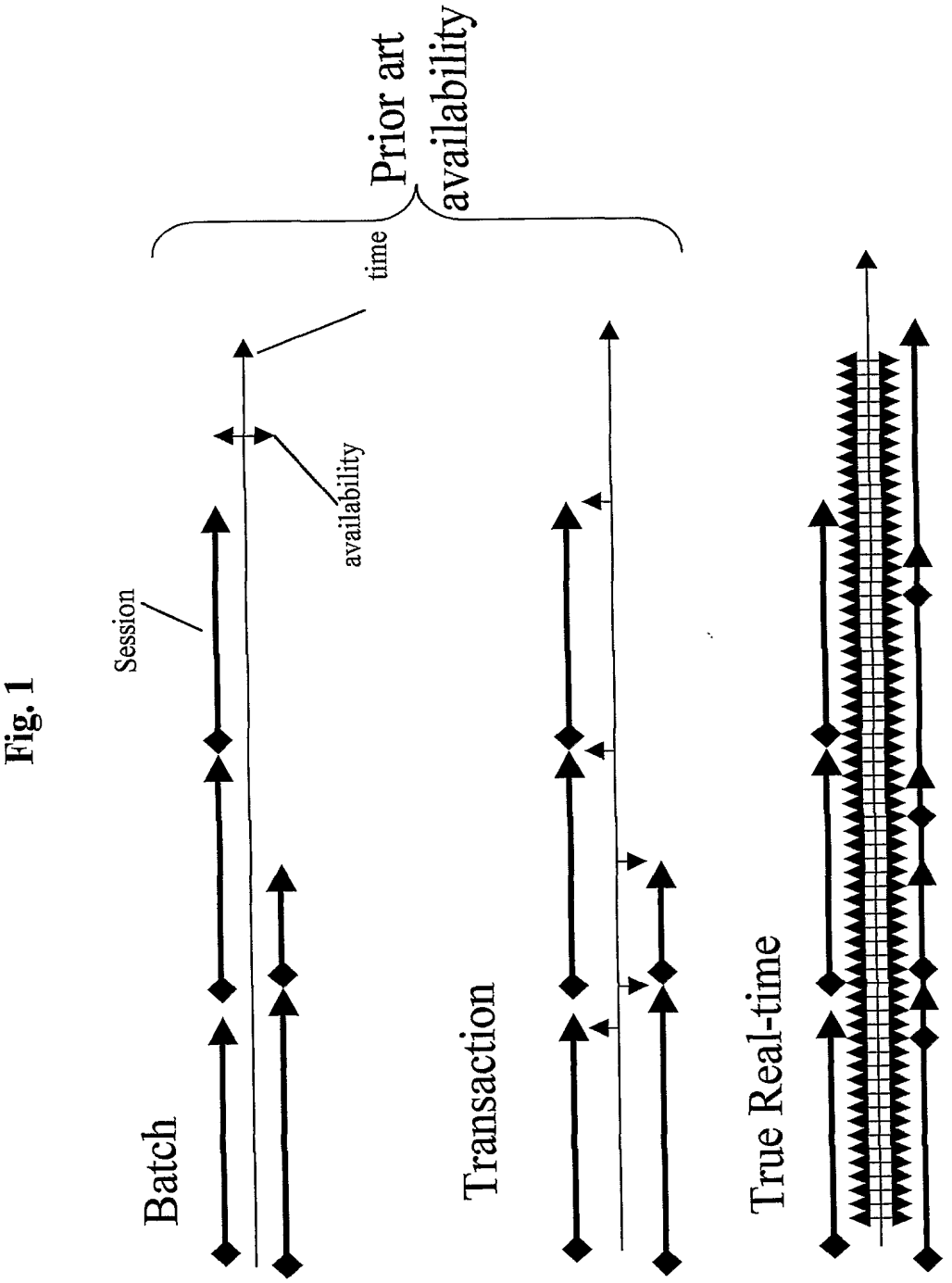
37. The system of claim 36 also comprising a rules engines which manages predefined rules to be triggered in a determined state.

38. The system of claim 36 also comprising an enforcement process that performs real-time actions against the system while a user session is active.

39. The system of claim 28 wherein the server is capable of providing the value of a parameter at any given moment in time.

36. The system of claim 28 wherein the server stores a parameter relating to a user as a function of current time.

Fig. 1


Prior art availability
time
Session
availability
Batch
Transaction
True Real-time

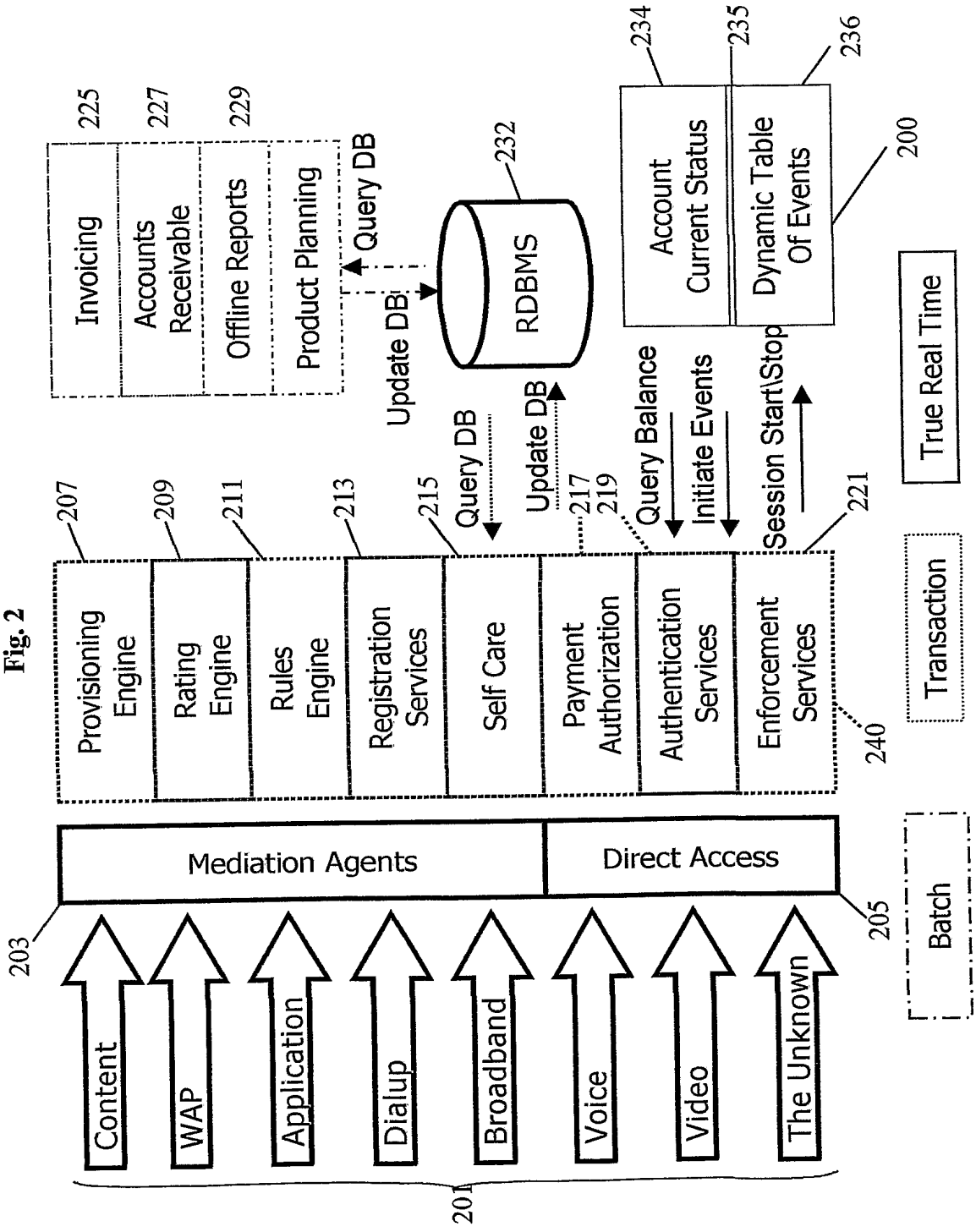

Content
WAP
Application
Dialup
Broadband
Voice
Video
The Unknown
201
203
Mediation Agents
Direct Access
205
Batch
Transaction
True Real Time
Provisioning Engine
Rating Engine
Rules Engine
Registration Services
Self Care
Payment Authorization
Authentication Services
Enforcement Services
207
209
211
213
215
217
219
221
240
Query DB
Update DB
Query Balance
Initiate Events
Session Start\Stop
Invoicing
Accounts Receivable
Offline Reports
Product Planning
225
227
229
Query DB
Update DB
RDBMS
232
Account Current Status
Dynamic Table Of Events
234
235
236
200

Fig. 2

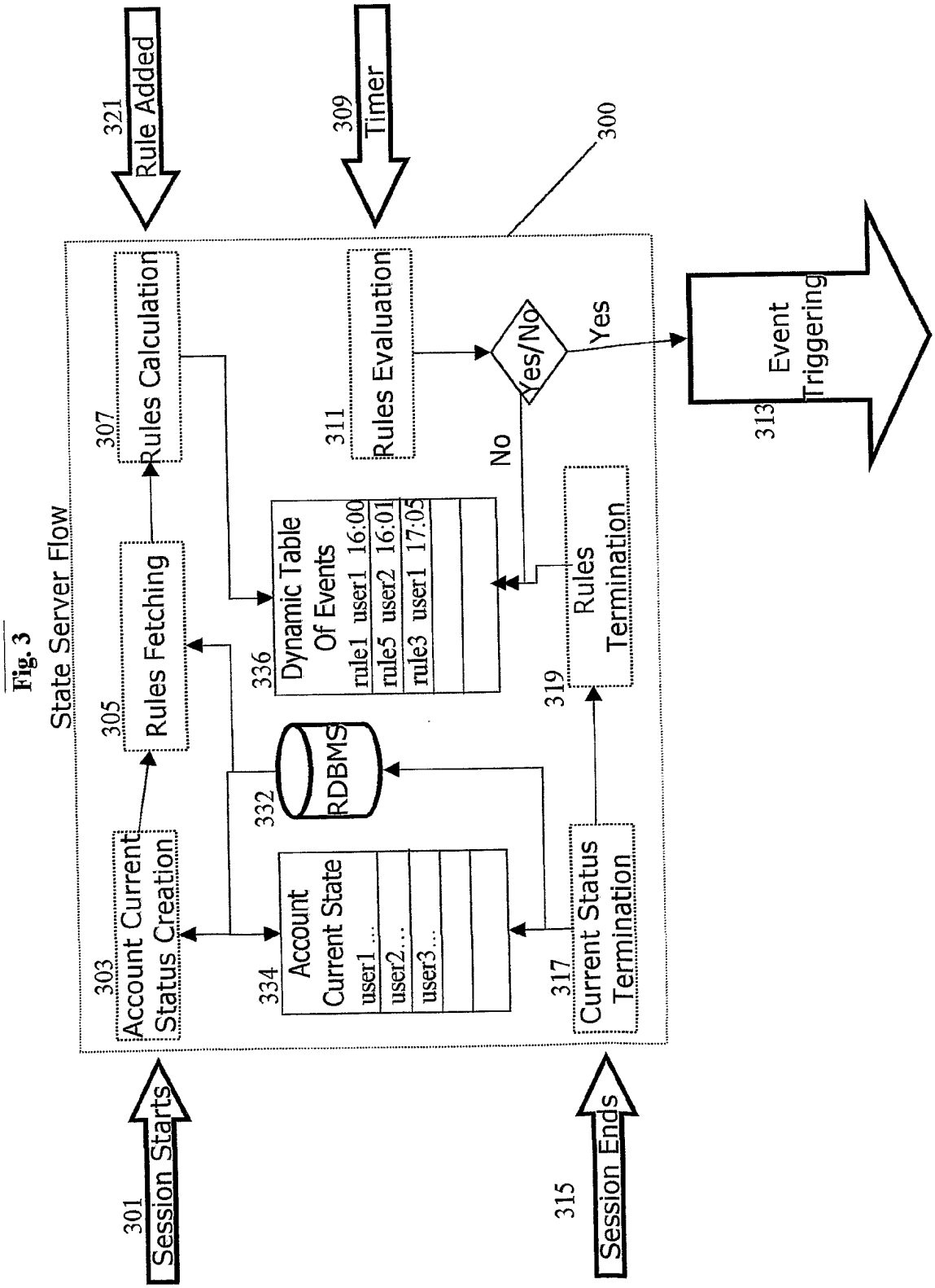

State Server Flow
300
301
Session Starts
303
Account Current Status Creation
305
Rules Fetching
307
Rules Calculation
309
Timer
311
Rules Evaluation
313
Event Triggering
315
Session Ends
317
Current Status Termination
319
Rules Termination
321
Rule Added
332
RDBMS
334
Account Current State
user1 ...
user2...
user3...
336
Dynamic Table Of Events
rule1 user1 16:00
rule5 user2 16:01
rule3 user1 17:05
Yes/No
Yes
No

Fig. 3